Towards Generalized and Efficient Metric Learning on Riemannian Manifold

Pengfei Zhu, Hao Cheng, Qinghua Hu*, Qilong Wang, Changqing Zhang
School of Computer Science and Technology, Tianjin University, Tianjin 300350, China
zhupengfei@tju.edu.cn, huqinghua@tju.edu.cn

Abstract

Modeling data as points on non-linear Riemannian manifolds has attracted increasing attentions in many computer vision tasks, especially visual recognition. Learning an appropriate metric on Riemannian manifold plays a key role in achieving promising performance. For widely used symmetric positive definite (SPD) manifold and Grassmann manifold, most of existing metric learning methods are designed for one manifold, and are not straightforward for the other one. Furthermore, optimizations in previous methods usually rely on computationally expensive iterations. To address above limitations, this paper makes an attempt to propose a generalized and efficient Riemannian manifold metric learning (RMML) method, which can be flexibly adopted to both SPD and Grassmann manifolds. By minimizing the geodesic distance of similar pairs and the interpoint geodesic distance of dissimilar ones on nonlinear manifolds, the proposed RMML is optimized by computing the geodesic mean between inverse of similarity matrix and dissimilarity matrix, benefiting a global closed-form solution and high efficiency. The experiments are conducted on various visual recognition tasks, and the results demonstrate our RMML performs favorably against its counterparts in terms of both accuracy and efficiency.

1 Introduction

Many works on computer vision tasks (*e.g.*, visual tracking, action recognition, and motion segmentation) have shown methods built on non-linear manifolds are superior to ones in Euclidean space. Among them, Riemannian manifold of symmetric positive definite (SPD) matrices [Pennec *et al.*, 2006; Tuzel *et al.*, 2008; Caseiro *et al.*, 2011; Harandi *et al.*, 2012; Li *et al.*, 2013; Caseiro *et al.*, 2013; Huang *et al.*, 2015b; Jayasumana *et al.*, 2015; Wang *et al.*, 2016; Harandi *et al.*, 2017] and Grassmann manifold [Hamm and Lee, 2008; Harandi *et al.*, 2011; Turaga *et al.*, 2011; Harandi *et al.*, 2014b; Huang *et al.*, 2015a; Jayasumana *et al.*, 2015] are two most widely used manifolds in computer vision community. The SPD matrices capture high order statistics of data, and have several different forms, *e.g.*, covariance region descriptors of images [Tuzel *et al.*, 2008], covariance matrices of image sets [Wang *et al.*, 2012], resulting matrix of Gaussian distribution [Wang *et al.*, 2017], and structure tensors [Caseiro *et al.*, 2011]. Grassmann manifold is a special type of Riemannian manifold, which is composed of sets of linear subspaces with the same dimensionality in Euclidean space [Jayasumana *et al.*, 2015; Huang *et al.*, 2015a].

Existing metric learning methods on SPD manifold and Grassmann manifold can be roughly divided into three categories in terms of projection space, *i.e.*, from manifold to tangent space [Li *et al.*, 2013; Huang *et al.*, 2015b], from manifold to Reproducing Kernel Hilbert Space (RKHS) [Hamm and Lee, 2008; Harandi *et al.*, 2011; 2012; Jayasumana *et al.*, 2015] and from manifold to low-dimensional manifold [Huang *et al.*, 2015a; Harandi *et al.*, 2017]. The methods based on projection of tangent space make traditional Euclidean methods workable on non-linear manifolds, but they do not take full advantage of geometry structures of manifolds. The methods projecting data points on manifolds into RKHS via some kernel functions seem make sense, however they are high computational cost and not scalable to large scale problems. In addition, some methods aim at learning metric to project original manifold to a more discriminative manifold [Harandi *et al.*, 2017; Huang *et al.*, 2015a] with full consideration of geometry structure of manifold. Although various metric learning methods have been studied on nonlinear manifolds, aforementioned methods are only designed for a specific type of manifold. Furthermore, most existing methods rely on optimization problems requiring computationally expensive iterations, where complex Riemannian operations usually are performed.

Towards addressing above problems, we propose a generalized and efficient Riemannian manifold metric learning (RMML) method, which is adopted on both SPD manifold and Grassmann manifold. Inspired by Euclidean metric learning method [Zadeh *et al.*, 2016], our manifold metric learning method is formulated as seeking a transformation matrix to minimize sum of geodesic distances of all similar pairs and the interpoint geodesic distances of all dissimilar ones on either SPD manifold or Grassmann manifold. It is both strict-

*The corresponding author

ly convex and strictly geodesically convex. To optimize our RMML, we first demonstrate the similar and dissimilar matrices associated with data points on manifolds are symmetric positive semi-definite, then the global closed-form solution of our proposed RMML can be achieved by computing matrix geometric mean between inverse of similarity matrix and dissimilarity matrix. It is worth mentioning that, instead of costly Riemannian geometric mean in [Zadeh *et al.*, 2016], we compute matrix geometric mean using more efficient Log-Euclidean Riemannian framework [Arsigny *et al.*, 2007] to further improve efficiency. The experimental results on image set classification, video based face recognition, and material classification tasks demonstrate the favorable performance and high efficiency of our RMML method against the state-of-the-art counterparts.

2 Related Work

Metric learning on SPD manifold. Affine Invariant Riemannian metric (AIRM) [Pennec *et al.*, 2006] and Log-Euclidean Riemannian metric (LERM) [Arsigny *et al.*, 2007] are widely used to handle SPD matrices. The LERM framework maps SPD manifold to its tangent space (a linear space) through the matrix logarithm, performing more efficient than AIRM framework, so many methods on SPD manifold are developed based on LERM. To learn the metric for $d \times d$ SPD matrices, methods in [Sivalingam *et al.*, 2009; Carreira *et al.*, 2012; Vemulapalli and Jacobs, 2015] firstly embed an $d \times d$ SPD matrix into its tangent space (a $d(d+1)/2$ dimensional linear space) under LERM framework, then perform classical Euclidean metric learning methods. Instead of learning metric in tangent space, methods in [Huang *et al.*, 2015b; Harandi *et al.*, 2017] learn intrinsic metrics to project original SPD manifold to a more discriminative SPD manifold characterized by $\mathbf{M}\mathbf{\Sigma}\mathbf{M}^T$ with a transformation matrix $\mathbf{M}$ for SPD matrix $\mathbf{\Sigma}$. In contrast to above methods, kernel-based methods [Wang *et al.*, 2012; Quang *et al.*, 2014] exploit some kernel functions to map SPD matrices into a reproducing kernel Hilbert space (RKHS), where kernelized methods (e.g., linear discriminant analysis (kLDA) and partial least square (kPLS)) are performed for discriminative learning.

Metric learning on Grassmann manifold. Compared to metric learning methods on SPD manifold, metric learning on Grassmann manifold are performed in either RKHS or the original manifold. For kernel-based metric learning methods, kernel functions are derived on Grassmann manifold to map original space to a high-dimensional RKHS [Hamm and Lee, 2008; Harandi *et al.*, 2011; 2014b; Jayasumana *et al.*, 2015]. Therein, Grassmann Discriminant Analysis (GDA) uses Fisher criterion to learn a low-dimensional space mapping [Hamm and Lee, 2008]. Grassmann Graph-embedding Discriminant Analysis (GGDA) exploits the graph embedding strategy to further improve the performance of GDA [Harandi *et al.*, 2011]. In [Jayasumana *et al.*, 2015], projection Gaussian kernel is derived and various applications in RKHS are discussed. Different from kernel-based methods, [Huang *et al.*, 2015b] learns a discriminative projection metric on the original Grassmann manifold.

The formulation of the proposed RMML is quite different from existing ones on Riemannian manifold. Our RMML method can learn intrinsic metrics on both SPD and Grassmann manifolds in the same formulation, which takes into full consideration geometry structures of manifolds. Moreover, its optimization can be achieved by a global closed-form solution benefiting high efficiency.

3 Parametric Riemannian Manifold Metrics

In this section, we first give a brief introduction of parametric metrics on SPD and Grassmann manifolds, which will be adopted to our proposed RMML method.

3.1 Parametric Metrics on SPD Manifold

We use $(\mathcal{M}, d)$ to represent a Riemannian manifold, where $\mathcal{M}$ is the topological space and d is the dimension of the manifold. Notably, Riemannian metric is a family of inner products on all tangent spaces and $(\mathcal{M}, d)$ is a differentiable manifold equipped with a smoothly varying inner product on each tangent space. It is well known that space of $d \times d$ SPD matrices forms a Riemannian manifold endowed with a Riemannian metric, called SPD manifold S_+^d. The LERM and AIRM are two most commonly used metrics to match SPD matrices. Although AIRM has better properties[1], it suffers from high computational complexity [Arsigny *et al.*, 2007]. Therefore, we exploit LERM in this paper.

For a point $\mathbf{S}_1 \in S_+^d$, the set of all tangent vectors at $\mathbf{S}_1$ on S_+^d constructs the tangent space of $\mathbf{S}_1$, denoted as $T_{\mathbf{S}_1} S_+^d$. Log-Euclidean metric exploits the Lie group structure of SPD matrices, benefiting low computational cost [Arsigny *et al.*, 2007]. For two points $\mathbf{S}_1$, $\mathbf{S}_2$ on the SPD manifold S_+^d, the logarithmic multiplication is defined as:

$$\mathbf{S}_1 \odot \mathbf{S}_2 = \exp\left(\log\left(\mathbf{S}_1\right) + \log\left(\mathbf{S}_2\right)\right), \tag{1}$$

where $\log(\cdot)$ and $\exp(\cdot)$ are matrix logarithmic and exponential operations, respectively. Under LERM framework, exponential and logarithmic maps can be represented in the forms of matrix exponential and logarithmic operations, *i.e.*,

$$\begin{cases} \exp_{\mathbf{S}_1}(\mathbf{L}) = \exp(\log(\mathbf{S}_1) + D_{\mathbf{S}_1} \log .\mathbf{L}). \\ \log_{\mathbf{S}_1}(\mathbf{S}_2) = D_{\log(\mathbf{S}_1)} \exp .(\log(\mathbf{S}_2) - \log(\mathbf{S}_1)). \end{cases} \tag{2}$$

Then, the scalar product between two elements $\mathbf{V}_1$ and $\mathbf{V}_2$ at a point $\mathbf{S}$ is calculated as

$$\langle \mathbf{V}_1, \mathbf{V}_2 \rangle_{\mathbf{S}} = \langle D_{\mathbf{S}} \log .\mathbf{V}_1, D_{\mathbf{S}} \log .\mathbf{V}_2 \rangle . \tag{3}$$

After that, the geodesic distance between two SPD matrices is defined as:

$$d_g(\mathbf{S}_1, \mathbf{S}_2) = \left\|\log\left(\mathbf{S}_1\right) - \log\left(\mathbf{S}_2\right)\right\|_F^2 . \tag{4}$$

In this way, the geodesic distance on SPD manifold is converted to Euclidean distance in the tangent space at identity matrix with matrix logarithm. Please refer to [Arsigny *et al.*, 2007] for more details.

In our applications, we first compute SPD matrices for a set of images or features (denoted as $\mathbf{S} = \{\mathbf{S}_1, \mathbf{S}_2, ..., \mathbf{S}_n\}$). Let $f : S_+^d \to S_+^r$ be a smooth mapping from original SPD

[1] AIRM is affine invariant, while LERM is similarity invariant.

manifold to a new manifold S_+^r. Thus, for a point $\mathbf{S}_i$, the mapping of tangent space from S_+^d to S_+^r is defined as:

$$TF\left(\mathbf{S}_i\right): T_{\mathbf{S}_i}S_+^d \to T_{f(\mathbf{S}_i)}S_+^r, \tag{5}$$

where the mapping $TF\left(\mathbf{S}_i\right)$ is an injection and mapping f is a smooth map [Huang *et al.*, 2015b]. Thus if we can find a transformation $\mathbf{M} \in R^{d\times r}$ of point $\mathbf{S}_i$ from original tangent space $T_{\mathbf{S}_i}S_+^d$ to space $T_{f(\mathbf{S}_i)}S_+^r$, geodesic distance between $\mathbf{S}_i$,$\mathbf{S}_j$ on original SPD manifold can be represented as:

$$d_g\left(\mathbf{S}_i, \mathbf{S}_j\right) = \left\|\mathbf{M}^T \log\left(\mathbf{S}_i\right)\mathbf{M} - \mathbf{M}^T \log\left(\mathbf{S}_j\right)\mathbf{M}\right\|_F^2, \tag{6}$$

where $\mathbf{MM^T}$ is a rank-r symmetric positive semi-definite (SPSD) matrix ensuring that the converted space is a tangent space of SPD matrices in the logarithm domain. Let $\mathbf{T} = \log\left(\mathbf{S}\right)$, and $\mathbf{Q} = \mathbf{MM}^T$. We can rewrite the formulation (6) as:

$$\begin{aligned} d_g\left(\mathbf{T}_i, \mathbf{T}_j\right) &= tr((\mathbf{T}_i - \mathbf{T}_j)^T \mathbf{Q}\left(\mathbf{T}_i - \mathbf{T}_j\right)\mathbf{Q}) \\ &= tr\left(\mathbf{A}\left(\mathbf{T}_i - \mathbf{T}_j\right)\left(\mathbf{T}_i - \mathbf{T}_j\right)\right), \end{aligned} \tag{7}$$

where $\mathbf{Q}$ is a rank-r SPSD matrix, so $\mathbf{A}{=}\mathbf{QQ}$ is also a rank-r SPSD matrix. Given an SPD matrix $\mathbf{S}$, we first embed it to $\mathbf{T}$ with mapping function ϕ, e.g., matrix logarithm or square root operation [Wang *et al.*, 2017], then compare two SPD matrices using Eq. (7). We can see that metric in Eq. (7) involves the parameter $\mathbf{A}$, which will be learned for more discriminative matching or classification.

3.2 Parametric Metrics on Grassmann Manifold

Let $\mathcal{G}$ be an n-dimensional vector space, a Grassmann manifold $Gr(k, \mathcal{G})$ can be regarded as a set of all k-dimensional subspaces in $\mathcal{G}$, denoted as $Gr(k, d)(k < d)$. An element on Grassmann manifold $Gr(k, d)$ is a linear subspace spanned by a $d \times k$ full rank orthonormal basis matrix $\mathbf{Y}$, where $\mathbf{Y}^T\mathbf{Y} = \mathbf{I}_k$ and $\mathbf{I}_k$ is an identity matrix of the size $k \times k$. In this paper, we employ commonly used projection distance to match two points $\mathbf{Y}_1$, $\mathbf{Y}_2$ on Grassmann manifold, *i.e.*,

$$d_p(\mathbf{Y}_1, \mathbf{Y}_2) = \left\|\mathbf{Y}_1\mathbf{Y}_1^T - \mathbf{Y}_2\mathbf{Y}_2^T\right\|_F, \tag{8}$$

where ${\mathbf{Y}_1\mathbf{Y}_1}^T$ and ${\mathbf{Y}_2\mathbf{Y}_2}^T$ are projection matrices and above projection distance approaches the geodesic distance at a scale of $\sqrt{2}$ [Huang *et al.*, 2015a].

Suppose there is a set of images or features $\{\mathbf{X}_1, \mathbf{X}_2, ..., \mathbf{X}_n\}$, where $\mathbf{X}_i \in \mathbb{R}^{d\times m_i}$, d is the size of sample and m_i is the number of samples in the i^{th} target, $i = 1, 2, .., n$. Firstly, we represent data $\mathbf{X}_i$ using a k-dimensional linear space, which is spanned by the corresponding orthonormal basis matrix, *i.e.*, $\mathbf{X}_i\mathbf{X}_i^T \simeq \mathbf{Y}_i\mathbf{\Lambda}_i\mathbf{Y}_i^T$, where $\mathbf{Y}_i$ is the matrix composed of the top k largest eigenvectors of $\mathbf{X}_i$. Then the distance between $\mathbf{X}_i$ and $\mathbf{X}_j$ can be converted to the distance between $\mathbf{Y}_i\mathbf{Y}_i^T$ and $\mathbf{Y}_j\mathbf{Y}_j^T$. Similar to the distance metrics on SPD manifold, we first define a mapping $f: Gr\left(k, d\right) \to Gr\left(k, r\right)$, and denote the transformation matrix of f as $\mathbf{M}$. Notably, $\mathbf{M}$ is a $d \times r$ matrix of full column rank. We use $\mathbf{M}^T{\mathbf{Y}_i}'$ to represent the orthonormal components of $\mathbf{M}^T\mathbf{Y}_i$, which is the orthonormal basis matrix of the new Grassmann manifold $Gr\left(k, r\right)$. Then the learned projection metric on the low-dimensional Grassmann manifold is defined as

$$\begin{aligned} & d_p^2\left(\mathbf{M}^T{\mathbf{Y}_i}'{\mathbf{Y}_i}'^T\mathbf{M}, \mathbf{M}^T{\mathbf{Y}_j}'{\mathbf{Y}_j}'^T\mathbf{M}\right) \\ &= 2^{-1/2}\left\|\mathbf{M}^T{\mathbf{Y}_i}'{\mathbf{Y}_i}'^T\mathbf{M} - \mathbf{M}^T{\mathbf{Y}_j}'{\mathbf{Y}_j}'^T\mathbf{M}\right\|_F^2 \\ &= 2^{-1/2} tr\left(\mathbf{QT}_{ij}\mathbf{T}_{ij}^T\mathbf{Q}\right), \end{aligned} \tag{9}$$

where $\mathbf{Q} = \mathbf{MM}^T$ is a $d \times d$ rank-r SPSD matrix and $\mathbf{T}_{ij} = {\mathbf{Y}_i}'{\mathbf{Y}_i}'^T - {\mathbf{Y}_j}'{\mathbf{Y}_j}'^T$. Let $\mathbf{A} = \mathbf{QQ}$, then we can rewrite the projection distance (9) as:

$$d_p\left(\mathbf{T}_{ij}\right) = tr\left(\mathbf{AT}_{ij}\mathbf{T}_{ij}\right), \tag{10}$$

where $\mathbf{A}$ is a parameter to be learned.

Notably, we find that parametric metric on Grassmann manifold (10) shares the similar form with one on SPD manifold (7), which encourages us to develop a metric learning method to handle both manifolds.

4 Riemannian Manifold Metric Learning

In this section, we propose a generalized metric learning method. The corresponding model and optimization are described as follows.

4.1 Model

Inspired by the geometric mean metric learning (GMML) in Euclidean space [Zadeh *et al.*, 2016], we present a novel metric learning method to learn the distance metrics for both SPD manifold and Grassmann manifold. Given a set of training data $\{\mathbf{X}_1, ...\mathbf{X}_i, ..., \mathbf{X}_n\}$, $\mathbf{X}_i \in \mathbb{R}^{d\times m_i}$ where d and m_i are feature dimension and the number of samples in the i^{th} target, respectively. In real-world applications, $\mathbf{X}_i$ can be an image set in video based face recognition, or a set of local features extracted from images. The goal of our method is to learn an appropriate parameter $\mathbf{A}$ in metrics (7) and (10) to further improve performance of model matching.

According to training data $\{\mathbf{X}_1, ...\mathbf{X}_i, ..., \mathbf{X}_n\}$ with label information, we can compute a set of SPD matrices $\{\mathbf{S}_1, ...\mathbf{S}_i, ..., \mathbf{S}_n\}$ with $\mathbf{S}_i \in \mathbb{R}^{d\times d}$ or a set of linear subspaces $\{\mathbf{Y}_1, ...\mathbf{Y}_i, ..., \mathbf{Y}_n\}$ with $\mathbf{Y}_i \in \mathbb{R}^{k\times d}$ on SPD manifold or Grassmann manifold. By using available label information, we can generate a set of positive pairs (two points with the same label) and negative pairs (two points with different labels). The principle of metric learning is usually to learn parameter (i.e., $\mathbf{A}$ in our case) to pull the distances between positive (similar) pairs ($(h_i, h_j) \in \mathcal{P}$) to be as small as possible, while the distances between negative (dissimilar) pairs ($(h_i, h_j) \in \mathcal{N}$) are pushed as large as possible.

As suggested in [Zadeh *et al.*, 2016], instead of decreasing the sum of distances over all similar pairs while increasing ones over all dissimilar pairs in previous methods, we learn $\mathbf{A}$ with minimizing the geodesic distance of similar pairs and the interpoint geodesic distance (described by $\mathbf{A}^{-1}$) of dissimilar ones on nonlinear manifolds. Then our metric learning method can be formulated as:

$$\min_{\mathbf{A}\succ 0} \sum_{(h_i,h_j)\in\mathcal{P}} d_{\mathbf{A}}(h_i, h_j) + \sum_{(h_i,h_j)\in\mathcal{N}} d_{\mathbf{A}^{-1}}(h_i, h_j), \tag{11}$$

where h_i and h_j represent two points on SPD manifold or Grassmann manifold. As shown in [Zadeh *et al.*, 2016], both of distances $d_{\mathbf{A}}$ and $d_{\mathbf{A}^{-1}}$ are monotonous but their tendencies of change are completely opposite. That is, $\sum_{(h_i,h_j)\in\mathcal{P}} d_{\mathbf{A}}(h_i,h_j)$ is minimized to pull the points in positive pairs closer while $\sum_{(h_i,h_j)\in\mathcal{N}} d_{\mathbf{A}^{-1}}(h_i,h_j)$ is minimized to enlarge the distances between two points in negative pairs.

According to the distance metrics in (7) and (10), our metric learning problem are described as follows. For SPD manifold, the objective function is formulated as:

$$\min_{\mathbf{A}\succ 0}\left\{\begin{array}{l}\sum\limits_{(h_i,h_j)\in\mathcal{P}} tr\left(\mathbf{A}\left(\mathbf{T}_i-\mathbf{T}_j\right)\left(\mathbf{T}_i-\mathbf{T}_j\right)\right)+\\ \sum\limits_{(h_i,h_j)\in\mathcal{N}} tr\left(\mathbf{A}^{-1}\left(\mathbf{T}_i-\mathbf{T}_j\right)\left(\mathbf{T}_i-\mathbf{T}_j\right)\right)\end{array}\right\}, \tag{12}$$

where $\mathbf{T}_i = log(\mathbf{S}_i)$ or $\mathbf{S}^{\frac{1}{2}}$, and $\mathbf{S}_i$ is an SPD matrix. Let $\mathbf{P}$ and $\mathbf{N}$ denote similar matrix and dissimilar matrix on the manifold, respectively. Thus, we have

$$\begin{array}{l}\mathbf{P}=\sum\limits_{(h_i,h_j)\in\mathcal{P}}\left(\mathbf{T}_i-\mathbf{T}_j\right)\left(\mathbf{T}_i-\mathbf{T}_j\right).\\ \mathbf{N}=\sum\limits_{(h_i,h_j)\in\mathcal{N}}\left(\mathbf{T}_i-\mathbf{T}_j\right)\left(\mathbf{T}_i-\mathbf{T}_j\right).\end{array} \tag{13}$$

For Grassmann manifold, the objective function is formulated as:

$$\min_{\mathbf{A}\succ 0}\left\{\sum_{(h_i,h_j)\in\mathcal{P}} tr\left(\mathbf{A}\mathbf{T}_{ij}\mathbf{T}_{ij}\right)+\sum_{(h_i,h_j)\in\mathcal{N}} tr\left(\mathbf{A}^{-1}\mathbf{T}_{ij}\mathbf{T}_{ij}\right)\right\}, \tag{14}$$

where $\mathbf{T}_{ij}=\mathbf{Y}_i{}'\mathbf{Y}_i{}'^T-\mathbf{Y}_j{}'\mathbf{Y}_j{}'^T$. Then, we have

$$\begin{array}{l}\mathbf{P}=\sum\limits_{(h_i,h_j)\in\mathcal{P}}\mathbf{T}_{ij}\mathbf{T}_{ij}.\\ \mathbf{N}=\sum\limits_{(h_i,h_j)\in\mathcal{N}}\mathbf{T}_{ij}\mathbf{T}_{ij}.\end{array} \tag{15}$$

We can see that our metric learning on both SPD (12) and Grassmann (15) manifolds can be reformulated in terms of $\mathbf{A}$, $\mathbf{P}$ and $\mathbf{N}$ as following:

$$f\left(\mathbf{A}\right)=\min_{\mathbf{A}\succ 0} tr\left(\mathbf{A}\mathbf{P}\right)+tr\left(\mathbf{A}^{-1}\mathbf{N}\right). \tag{16}$$

Meanwhile, we show $\mathbf{P}$ and $\mathbf{N}$ are positive semi-definite for both SPD manifold and Grassmann manifold, which is given in the following lemma:

Lemma 1 *For SPD manifold and Grassmann manifold,* $\mathbf{P}$ *and* $\mathbf{N}$ *in* (13) *and* (15) *are symmetric positive semi-definite (SPSD).*

Proof 1 *For two points* $\mathbf{S}_i, \mathbf{S}_j$ *on SPD manifold, the matrices* $\mathbf{T}_i, \mathbf{T}_j$ *are SPD matrices as well.* $\mathbf{T}_i-\mathbf{T}_j$ *is also a symmetric matrix. Since the square of an arbitrary symmetric real matrix is a SPSD matrix,* $\left(\mathbf{T}_i-\mathbf{T}_j\right)\left(\mathbf{T}_i-\mathbf{T}_j\right)$ *is easily proved to be a SPSD matrix. Hence,* $\mathbf{P}$ *and* $\mathbf{N}$ *are symmetric positive semi-definite.*

For Grassmann manifold, as $\mathbf{Y}_i{}'\mathbf{Y}_i{}'^T$ *is symmetric,* $\mathbf{T}_{ij}$ *is also symmetric. Thus, it is easily proved that for Grassmann manifold* $\mathbf{P}$ *and* $\mathbf{N}$ *are symmetric positive semi-definite.*

The property of positive semi-definite lying in $\mathbf{P}$ and $\mathbf{N}$ plays a key role in optimizing our RMML (16), which will be introduced as follows.

Algorithm 1 RMML-SPD and RMML-GM Algorithms

Input:
Training data $\left\{\mathbf{X}_i, \mathbf{X}_2, ..., \mathbf{X}_n\right\}, \mathbf{X}_i \in {}^{m_i\times d}$.
1: RMML-SPD: Compute SPD matrices $\mathbf{S}_i$ from $\mathbf{X}_i$, $i = 1, 2, ..., n$. Compute $\mathbf{T}_i$, and $\mathbf{P}$ and $\mathbf{N}$ by using (13).
RMML-GM: Compute $\mathbf{Y}_i$ for all train data $\mathbf{X}_i$ and $\mathbf{P}$ and $\mathbf{N}$ by using (15).
2: Compute $\mathbf{A}$ by using (21).
Output:
Distance Metric matrix $\mathbf{A}$

4.2 Optimization and Algorithm

Since objective function (16) is strictly convex and geodesically convex [Zadeh *et al.*, 2016], we can seek the globally optimal solution of $\mathbf{A}$ if it satisfies $\nabla f\left(\mathbf{A}\right)=0$. So, by computing the gradient of $\mathbf{f}$ with respect to $\mathbf{A}$ and setting it to zero, we have

$$\mathbf{P}=\mathbf{A}^{-1}\mathbf{N}\mathbf{A}^{-1}\Rightarrow\mathbf{A}\mathbf{P}\mathbf{A}=\mathbf{N}. \tag{17}$$

It is easy to know that (17) is a Riccati equation and its globally optimal solution is the midpoint of geodesic curve joining matrices $\mathbf{P}^{-1}$ and $\mathbf{N}$ on SPD manifold. So, the solution of $\mathbf{A}$ can be easily achieved by computing the Riemannian geometric mean between SPD matrices $\mathbf{P}^{-1}$ and $\mathbf{N}$, i.e,

$$\mathbf{A}=\mathbf{P}^{-1}\#_{1/2}\mathbf{N}=\mathbf{P}^{-1/2}\left(\mathbf{P}^{1/2}\mathbf{N}\mathbf{P}^{1/2}\right)^{1/2}\mathbf{P}^{-1/2}. \tag{18}$$

Obviously, since $\mathbf{A}$ is Riemannian geometric mean between SPD matrices $\mathbf{P}^{-1}$ and $\mathbf{N}$, it is naturally symmetric positive definite.

However, it is well known that computation of geometric mean (18) involves expensive Riemannian operations. In this paper, we optimize $\mathbf{A}$ as geometric mean between SPD matrices $\mathbf{P}^{-1}$ and $\mathbf{N}$ under efficient LERM framework. Thus, we have

$$\mathbf{A}=\exp\left(\frac{-\log(\mathbf{P})+\log(\mathbf{N})}{2}\right). \tag{19}$$

Compared with solution (18), our optimization (19) is more efficient. Note that solution (19) needs $\mathbf{P}^{-1}$ and $\mathbf{N}$ are symmetric positive definite. The Lemma 1 proves $\mathbf{P}^{-1}$ and $\mathbf{N}$ are naturally symmetric positive semi-definite, then we introduce a regularizer with parameter $\lambda \geq 0$ to ensure $\mathbf{P}$ and $\mathbf{N}$ be positive definite. In this way, the solution has

$$\mathbf{A}_R=\exp\left(\frac{-\log(\mathbf{P}+\lambda\mathbf{I})+\log(\mathbf{N}+\lambda\mathbf{I})}{2}\right). \tag{20}$$

Besides, a weight parameter t ($0 \leq t \leq 1$) is introduced to balance the effect of the distances between positive pairs and those between negative pairs. Thus, the final solution of matrix $\mathbf{A}$ is obtained as

$$\mathbf{A}_{final}=\exp\left(\frac{-t\log(\mathbf{P}+\lambda\mathbf{I})+(1-t)\log(\mathbf{N}+\lambda\mathbf{I})}{2}\right). \tag{21}$$

The proposed metric learning method on SPD manifold (RMML-SPD) and Grassmann manifold (RMML-GM) is summarized in Algorithm 1.

For each sample in probe sets, we compute the distance between it and each training sample with the matrix $\mathbf{A}$ computed in Algorithm 1, then we use simple NN for classification.

4.3 Discussions

Compared with pervious manifold metric learning methods (e.g., LEML [Huang *et al.*, 2015b] and PML [Huang *et al.*, 2015a]) relying on computationally expensive iterations, our RMML can be efficiently optimized with a global closed-form solution. Moreover, the learned parameter in our method is naturally positive definite without additional regularization. Our RMML is inspired by GMML [Zadeh *et al.*, 2016], however, the proposed RMML aims at learning metrics on non-linear manifolds rather than in linear Euclidean space. Furthermore, instead of optimization based on Riemannian operation in [Zadeh *et al.*, 2016], our RMML is optimized under more efficient LERM framework.

5 Experiments

In this section, we evaluate the proposed method on three tasks, including object recognition, video based face recognition and material classification.

Datasets. We conduct experiments on five datasets, including ETH-80 [Leibe and Schiele, 2003], Flickr Material dataset [Sharan *et al.*, 2009], and UIUC material [Liao *et al.*, 2013], YouTube Celebrities [Kim *et al.*, 2008], and YouTube Face dataset [Wolf *et al.*, 2011].

ETH-80 dataset contains 80 image sets of 8 object categories [Leibe and Schiele, 2003]. There are 10 sub-objects for each category, and each sub-object has 41 images from different views. Following the experimental settings in [Wang *et al.*, 2012], we randomly choose 5 objects as gallery and the other 5 objects as probes in each category. The size of each image is resized to 20×20, and the intensity feature is used.

Flickr material dataset (FMD) contains 1000 images of 10 materials categories in the wild [Sharan *et al.*, 2009]. We pass each image through VGG-VD16 model pre-trained on ImageNet dataset, and employ the outputs of the last convolution layer as local features with size of $m_i \times 512$. Following the common setting in [Wang *et al.*, 2016], we randomly choose half of images in each category as gallery and the other half as probes.

UIUC material dataset has 216 images and 18 categories of material in the wild [Liao *et al.*, 2013]. The feature extraction strategy is used as the same as that in Flickr material dataset. We randomly choose half of images in each category for gallery and the other half for probes.

YouTube Celebrities (YTC) dataset has 1910 video clips of 47 subjects [Kim *et al.*, 2008], and the number of frames in each video varies significantly. Following the common setting in [Wang *et al.*, 2012; Huang *et al.*, 2015b], we randomly select 3 image sets per subject for gallery and 6 image sets for probes. Histogram equalization is used to eliminate lighting effects as a preprocessing step. Then, each image is resized to a 20×20 image and intensity is used to represent images.

YouTube Face (YTF) dataset contains 3425 videos of 1595 peoples [Wolf *et al.*, 2011]. Each video has many sequences with variations of poses, illuminations and expressions. Following [Huang *et al.*, 2015b], we intercept the face part of every frame and resized it into 20×40 pixels. 5000 video pairs are used to perform ten-fold cross validation tests. In each fold there are 500 pairs, including 250 pairs of the same person and 250 pairs of different persons.

Evaluation methods. To evaluate the effectiveness and efficiency of our proposed method, we compare RMML with various state-of-the-art counterparts including nonlinear manifold learning methods and metric learning methods on manifolds. They are summarized as follows.

- Nonlinear manifold based methods: Manifold-Manifold Distance (MMD) [Wang *et al.*, 2008]; Manifold Discriminant Analysis (MDA) [Wang and Chen, 2009].
- Affine subspace based methods: Affine and Convex Hull based Image Set Distance (AHISD and CHISD) [Cevikalp and Triggs, 2010].
- SPD manifold based methods: SPD Manifold Learning (SPDML) [Harandi *et al.*, 2014a]; Log-Euclidean Metric Learning (LEML) [Huang *et al.*, 2015b].
- Grassmann manifold based methods: Grassmann Graph embedding Discriminant Analysis (GGDA) [Harandi *et al.*, 2011]. Projection Metric Learning (PML) [Huang *et al.*, 2015a];

Parameters setting. For metric learning on SPD manifold, we first compute mean vector μ and sample covariance $\mathbf{S}$ of a set of data to obtain a Gaussian descriptor. Meanwhile as suggested in [Wang *et al.*, 2016], we estimate covariance matrix $\hat{\Sigma}$ with vN-MLE method. As the space of Gaussian distribution forms a Riemannian manifold, we embed them into the space of SPD matrices by $\begin{bmatrix} \hat{\Sigma} + \beta^2\mu\mu^T & \beta\mu \\ \beta\mu^T & 1 \end{bmatrix}$, where $\beta > 0$ is a positive parameter to balance the dimension and orders of magnitude between mean vector and covariance.

For LEML method, since the distance metrics are learned in the tangent space, we compute $\mathbf{T}_i = \log(\mathbf{S}_i)$ for all SPD matrices. As suggested in [Huang *et al.*, 2015b], we add a positive real number (i.e.,$0.001 \times tr(\mathbf{S}_i)$) to the diagonal elements of $\mathbf{S}_i$ for numerical stability, and exploit Gaussian embedding followed by matrix logarithm. Following the existing metric learning algorithms on Grassmann manifold [Harandi *et al.*, 2011; Huang *et al.*, 2015a], the number of orthonormal bases for each linear subspace is set as 10.

For fair comparison, we exploit the source codes of competitive methods provided by the authors, and set the parameters suggested by the original papers. For MMD, the variance percentage to preserve in PCA is set to 90%. For MDA, we set three parameters as same as [Wang and Chen, 2009]. For linear and non-linear versions of AHISD and CHISD [Cevikalp and Triggs, 2010], the 98% energy by PCA is retained in non-linear AHISD and the value of error penalty C in CHISD is set as same as [Cevikalp and Triggs, 2010]. The reduced feature dimension is set to $(c-1)$ for LDA, where c is the number of classes. For LEML, η is tuned from 0.001 to 1000 and the value of ζ is tuned from 0.1 to 1. For GGDA, the graph parameter v is set from 1 to 10 and the size of projection matrix r is set to $(c-1)$. Besides, its parameter β is tuned from $1e^2$ to $1e^6$. There are two parameters λ and t for RMML. We simply set λ to 0.1 on all datasets, and set t from $\{0.2, 0.4, 0.6, 0.8\}$ by cross-validation on the training set.

Accuracy. The performances of all competitive methods are evaluated in terms of both accuracy and efficiency. Table **??** shows accuracies of different methods on five datasets. Ten random trials are run on ETH-80, FMD, UIUC material and YTC datasets, then the average results are reported. For YTF, the standard ten-fold cross validation are used for evaluation. As GGDA cannot be applied to verification task, the result of GGDA on YTF dataset is not reported. From Table **??** one can see that our RMML-SPD and RMML-GM get better performance than their counterparts in both classification and verification tasks. Compared with metric learning methods on SPD manifold, our RMML-SPD significantly improves the second best LEML over 1.50%, 2.28%, 7.13%, 8.20%, 2.36% on five datasets, respectively. Compared with metric learning methods on Grassmann manifold, the proposed RMML-GM achieves the best results on four datasets except UIUC. The promising performance of our RMML maybe owe to effective objective function and global closed-form solution.

Methods	ETH-80	FMD	UIUC	YTC	YTF
MMD	85.75	60.60	62.78	69.60	65.60
MDA	87.75	63.50	67.13	64.72	66.28
AHISD(linear)	72.50	46.72	55.37	64.65	66.50
AHISD(non-linear)	72.00	46.72	55.37	66.58	67.40
CHISD(linear)	79.75	47.52	65.09	67.24	66.24
CHISD(non-linear)	72.50	63.90	65.65	68.09	67.60
SPDML-AIRM	90.75	63.42	62.00	67.50	62.16
SPDML-Stein	90.75	63.80	61.12	68.10	62.56
LEML	93.50	66.60	62.96	69.85	68.80
RMML-SPD	**95.00**	**68.88**	**70.09**	**78.05**	**71.16**
GGDA	87.75	68.26	74.26	65.99	N/A
PML	90.25	68.88	**76.94**	68.79	69.44
RMML-GM	**93.00**	**69.62**	76.48	**69.15**	**70.20**

Table 1: Accuracies of different methods on five datasets

Methods	ETH-80	FMD	UIUC	YTC	YTF
MDA	0.31	13.21	7.25	11.30	1859.64
LEML	9.57	710.01	182.97	28.10	2340.3
RMML-SPD	**0.06**	**0.11**	**0.11**	**0.06**	**0.48**
GGDA	1.43	108.83	6.02	3.42	N/A
PML	3.56	2988.3	93.69	26.20	14.54
RMML-GM	**0.06**	**0.06**	**0.10**	**0.06**	**0.47**

Table 2: Comparison of training time (s) on five datasets

Running time. Then we compare the running time of different methods. The experiments are run on a PC equipped with a single Intel(R) Core(TM) i7-6700 (3.40GHz). For our RMML-SPD and RMML-GM, both SPD matrices and linear subspaces can be computed offline, leading the computation of $\mathbf{P}$ and $\mathbf{N}$ in (13) and (15) can be regarded as a preprocessing step. Therefore, we report the time of solving (21) as the training time of RMML. For LEML, the computation of SPD matrices and logarithmic operation is considered as a preprocessing step and the time for the cyclic Bregman projection algorithm is recorded as the training time. For PML, we compute the linear subspaces before training and report the time of projection metric learning. As for GGDA, the Gram matrix is first computed as a preprocessing step. The time for computing within-class and between-class graph similarity matrices and eigen decomposition is recorded.

The training time of various methods on five datasets is listed in Table 2. From it we can see that RMML-SPD and RMML-GM are much faster than LEML and PML, respectively. It owes to RMML has a closed-form solution while LEML and PML both need several iterations for optimization. Compared with GGDA, RMML is more scalable to large scale problems. Besides, RMML is also suitable for online learning because $\mathbf{P}$ and $\mathbf{N}$ in (13) and (15) can be easily updated by adding the distance information of new pairs. Then distance metrics are learned efficiently with a closed-form solution. On the contrary, metric learning methods relying on iterative optimization will be time consuming.

6 Conclusions and Future Work

In this paper, we proposed a novel Riemannian manifold metric learning (RMML) towards handling two most widely used manifolds (i.e., SPD manifold and Grassmann manifold) in the same formulation. To this end, we first introduce parametric metrics on SPD and Grassmann manifolds, and formulate metric learning problem as minimizing the geodesic distance of similar pairs and the interpoint geodesic distance of dissimilar ones on manifolds. The proposed RMML has a global optimum and is efficiently optimized with a closed-form solution under LERM framework. Experiments on various computer vision tasks demonstrate the effectiveness and efficiency of the proposed RMML. In future, we will adopt RMML to more possible types of Riemannian manifolds and apply our method to more other tasks.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grants 61502332 and 61732011, Natural Science Foundation of Tianjin Under Grants 17JCZDJC30800.

References

[Arsigny *et al.*, 2007] Vincent Arsigny, Pierre Fillard, Xavier Pennec, and Nicholas Ayache. Geometric means in a novel vector space structure on symmetric positive-definite matrices. *SIAM journal on matrix analysis and applications*, 29(1):328–347, 2007.

[Carreira *et al.*, 2012] Joao Carreira, Rui Caseiro, Jorge Batista, and Cristian Sminchisescu. Semantic segmentation with second-order pooling. In *ECCV*, pages 430–443, 2012.

[Caseiro *et al.*, 2011] Rui Caseiro, João F Henriques, Pedro Martins, and Jorge Batista. A nonparametric Riemannian framework on tensor field with application to foreground segmentation. In *ICCV*, pages 1–8, 2011.

[Caseiro *et al.*, 2013] Rui Caseiro, Pedro Martins, João F Henriques, Fatima Silva Leite, and Jorge Batista. Rolling riemannian manifolds to solve the multi-class classification problem. In *CVPR*, pages 41–48, 2013.

[Cevikalp and Triggs, 2010] Hakan Cevikalp and Bill Triggs. Face recognition based on image sets. In *CVPR*, pages 2567–2573, 2010.

[Hamm and Lee, 2008] Jihun Hamm and Daniel D Lee. Grassmann discriminant analysis: a unifying view on subspace-based learning. In *ICML*, pages 376–383, 2008.

[Harandi *et al.*, 2011] Mehrtash T Harandi, Conrad Sanderson, Sareh Shirazi, and Brian C Lovell. Graph embedding discriminant analysis on Grassmannian manifolds for improved image set matching. In *CVPR*, pages 2705–2712, 2011.

[Harandi *et al.*, 2012] Mehrtash T Harandi, Conrad Sanderson, Richard Hartley, and Brian C Lovell. Sparse coding and dictionary learning for symmetric positive definite matrices: A kernel approach. In *ECCV*, pages 216–229. 2012.

[Harandi *et al.*, 2014a] Mehrtash T Harandi, Mathieu Salzmann, and Richard Hartley. From manifold to manifold: Geometry-aware dimensionality reduction for SPD matrices. In *ECCV*, pages 17–32, 2014.

[Harandi *et al.*, 2014b] Mehrtash T Harandi, Mathieu Salzmann, Sadeep Jayasumana, Richard Hartley, and Hongdong Li. Expanding the family of Grassmannian kernels: An embedding perspective. In *ECCV*, pages 408–423, 2014.

[Harandi *et al.*, 2017] Mehrtash T Harandi, Mathieu Salzmann, and Richard Hartley. Dimensionality reduction on SPD manifolds: The emergence of geometry-aware methods. *TPAMI*, 40(1):48–62, 2017.

[Huang *et al.*, 2015a] Zhiwu Huang, Ruiping Wang, Shiguang Shan, and Xilin Chen. Projection metric learning on Grassmann manifold with application to video based face recognition. In *CVPR*, pages 140–149, 2015.

[Huang *et al.*, 2015b] Zhiwu Huang, Ruiping Wang, Shiguang Shan, Xianqiu Li, and Xilin Chen. Log-Euclidean metric learning on symmetric positive definite manifold with application to image set classification. In *ICML*, pages 720–729, 2015.

[Jayasumana *et al.*, 2015] Sadeep Jayasumana, Richard Hartley, Mathieu Salzmann, Hongdong Li, and Mehrtash Harandi. Kernel methods on Riemannian manifolds with Gaussian RBF kernels. *TPAMI*, 37(12):2464–2477, 2015.

[Kim *et al.*, 2008] Minyoung Kim, Sanjiv Kumar, Vladimir Pavlovic, and Henry Rowley. Face tracking and recognition with visual constraints in real-world videos. In *CVPR*, pages 1–8, 2008.

[Leibe and Schiele, 2003] Bastian Leibe and Bernt Schiele. Analyzing appearance and contour based methods for object categorization. In *CVPR*, volume 2, pages 402–409, 2003.

[Li *et al.*, 2013] Peihua Li, Qilong Wang, Wangmeng Zuo, and Lei Zhang. Log-Euclidean kernels for sparse representation and dictionary learning. In *ICCV*, pages 1601–1608, 2013.

[Liao *et al.*, 2013] Zicheng Liao, Jason Rock, Yang Wang, and David Forsyth. Non-parametric filtering for geometric detail extraction and material representation. In *CVPR*, pages 963–970, 2013.

[Pennec *et al.*, 2006] Xavier Pennec, Pierre Fillard, and Nicholas Ayache. A Riemannian framework for tensor computing. *IJCV*, 66(1):41–66, 2006.

[Quang *et al.*, 2014] Minh Ha Quang, Marco San Biagio, and Vittorio Murino. Log-Hilbert-Schmidt metric between positive definite operators on Hilbert spaces. In *NIPS*, pages 388–396, 2014.

[Sharan *et al.*, 2009] Lavanya Sharan, Ruth Rosenholtz, and Edward H. Adelson. Material perception: What can you see in a brief glance? *Journal of Vision*, 9(8):784–784, 2009.

[Sivalingam *et al.*, 2009] Ravishankar Sivalingam, Vassilios Morellas, Daniel Boley, and Nikolaos Papanikolopoulos. Metric learning for semi-supervised clustering of region covariance descriptors. In *ICDSC*, pages 1–8, 2009.

[Turaga *et al.*, 2011] Pavan Turaga, Ashok Veeraraghavan, Anuj Srivastava, and Rama Chellappa. Statistical computations on Grassmann and Stiefel manifolds for image and video-based recognition. *TPAMI*, 33(11):2273–2286, 2011.

[Tuzel *et al.*, 2008] Oncel Tuzel, Fatih Porikli, and Peter Meer. Pedestrian detection via classification on Riemannian manifolds. *TPAMI*, 30(10):1713–27, 2008.

[Vemulapalli and Jacobs, 2015] Raviteja Vemulapalli and David W Jacobs. Riemannian metric learning for symmetric positive definite matrices. *arXiv preprint arXiv:1501.02393*, 2015.

[Wang and Chen, 2009] Ruiping Wang and Xilin Chen. Manifold discriminant analysis. In *CVPR*, pages 429–436, 2009.

[Wang *et al.*, 2008] Ruiping Wang, Shiguang Shan, Xilin Chen, and Wen Gao. Manifold-manifold distance with application to face recognition based on image set. In *CVPR*, pages 1–8, 2008.

[Wang *et al.*, 2012] Ruiping Wang, Huimin Guo, Larry S Davis, and Qionghai Dai. Covariance discriminative learning: A natural and efficient approach to image set classification. In *CVPR*, pages 2496–2503, 2012.

[Wang *et al.*, 2016] Qilong Wang, Peihua Li, Wangmeng Zuo, and Lei Zhang. RAID-G: Robust estimation of approximate infinite dimensional Gaussian with application to material recognition. In *CVPR*, pages 4433–4441, 2016.

[Wang *et al.*, 2017] Qilong Wang, Peihua Li, and Lei Zhang. G2DeNet: Global Gaussian distribution embedding network and its application to visual recognition. In *CVPR*, pages 2730–2739, 2017.

[Wolf *et al.*, 2011] Lior Wolf, Tal Hassner, and Itay Maoz. Face recognition in unconstrained videos with matched background similarity. In *CVPR*, pages 529–534, 2011.

[Zadeh *et al.*, 2016] Pourya Habib Zadeh, Reshad Hosseini, and Suvrit Sra. Geometric mean metric learning. In *ICML*, pages 2464–2471, 2016.